Chapter 4
Low Level Elements

4.1 Tapered Beam

The tapered beam element in ADAMS/WT was developed as a step in the automation of construction of tapered, twisted, flexible rotor blades (section 5.6). It is also used for building tapered, flexible towers (section 5.2).

The existing BEAM element in ADAMS[3] models a straight, untwisted beam with uniform stiffness properties. (Remember that in ADAMS, beams are considered as forces and have no associated inertial properties.) The new tapered beam element in ADAMS/WT models a beam with stiffness properties that vary linearly between the two end points. The elastic axis, i.e. the line of shear centers, is assumed to be straight between the two ends, whose cross-sections are parallel in the undeflected position. The cross-section principal axes of the far end may be rotated moderately about the x-axis of the near end.

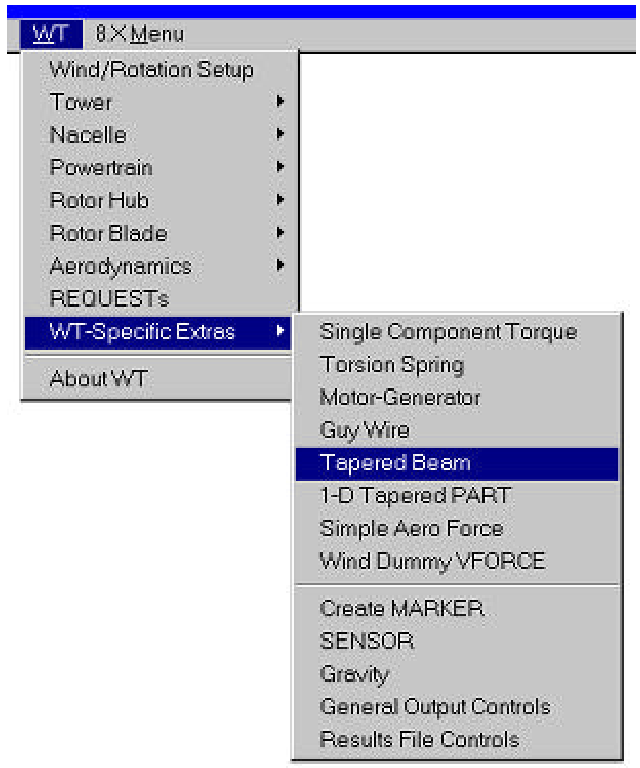
What appears as a tapered beam element in ADAMS/WT preprocessing is actually implemented in the model (and ADAMS dataset) as a specially-defined case of the existing ADAMS FIELD element. The FIELD element allows for an arbitrarily-defined 6x6 stiffness matrix relating the relative displacements of the two end MARKERs to the reaction forces at those ends. For the ADAMS/WT tapered beam, the terms in the FIELD's stiffness matrix were derived using classical beam theory and a finite element approach (see Bathe[4] for example). Cubic hermite polynomials were used for the lateral deflection shape functions, quadratics for the torsional and extensional deflection shape functions and a linear variation of stiffness properties was assumed. The appropriate stiffness integrals were evaluated symbolically and the definitions of the matrix terms coded directly into the ADAMS/WT macro which builds the tapered beam. A complete explanation and derivation of the tapered beam matrices can be found in Appendix D.

Note that in this element it is the combined <u>stiffness</u> properties, that is EI, GJ and EA, which are assumed to vary linearly over the beam's length, not the area properties nor the dimensions. Because of the typically complex internal construction of rotor blades and support towers, the assumption of linearly varying stiffness properties is likely to be closer to reality than either of the other assumptions.

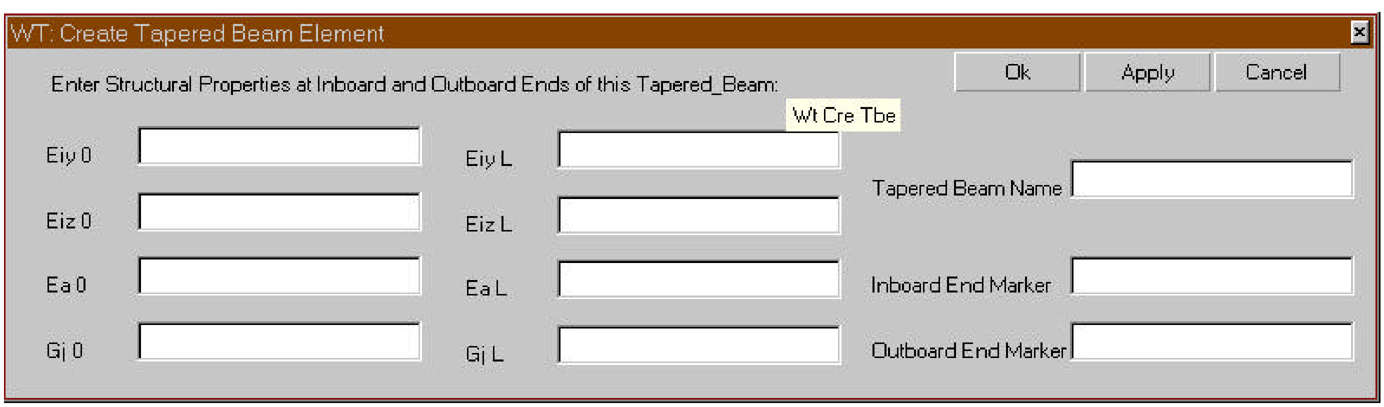
3. *Using ADAMS / Solver*, Version 9, "Statements", PN 91SOLVUG-01, Mechanical Dynamics, Inc, Ann Arbor, MI, 1998.

4.Bathe, K-J., *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, New Jersey, 1982.

As stated, tapered beam elements are normally generated automatically during creation of flexible rotor blades or towers. It is possible, however, to manually create a tapered beam element through the WT-Specific Extras menu:



The data entry panel for the tapered beam is shown below:



EIy_0

Chordwise stiffness at inboard end, N-m^2.

EIz_0

Flatwise stiffness at inboard end, N-m2.

EA_0

Extensional stiffness at inboard end, N.

GJ_0

Torsional stiffness at inboard end, N-m2.

EIy_L

Chordwise stiffness at outboard end, N-m2.

EIz_L

Flatwise stiffness at outboard end, N-m^2.

EA_L

Extensional stiffness at outboard end, N.

GJ_L

Torsional stiffness at outboard end, N-m2.

Tapered_Beam Name

ADAMS/View name for this element. ADAMS/WT uses "tbe_" as the prefix for automatically created beams to allow for bulk modifications to an aggregate element.

Inboard End Marker

ADAMS/View name for the MARKER at the inboard end (J marker) of the element. This marker also defines the reference coordinate for the computation of FIELD forces.

Outboard End Marker

ADAMS/View name for the MARKER at the outboard end (I marker) of the element.

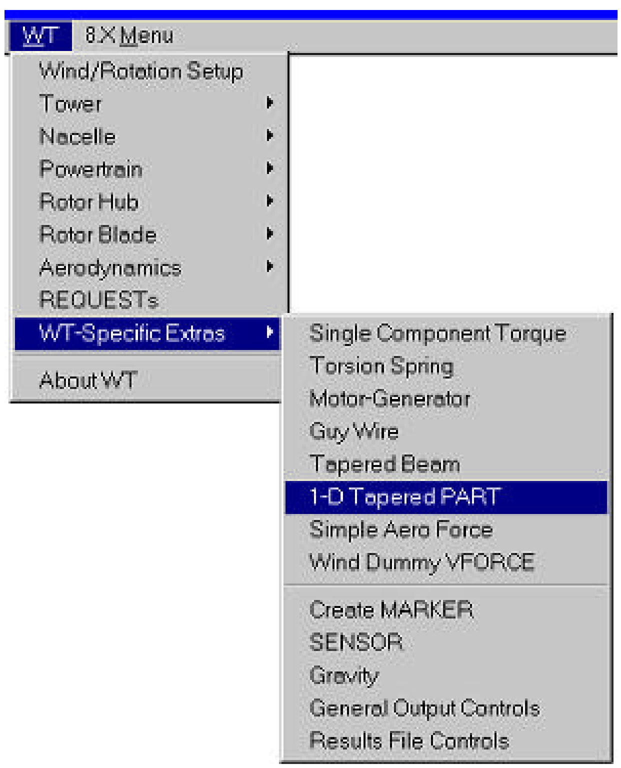
After all the data are entered and the panel is executed (by clicking on OK), ADAMS/WT will automatically run the macro code which computes the correct matrix terms and generates the desired FIELD element. For the flexible blade and tower aggregate elements, ADAMS/WT nests macros, repetitively accessing the tapered beam and tapered part (section 4.2) macros to build up the longer structures.

4.2 Tapered Part

The tapered part element in ADAMS/WT was also developed as a step in the automation of construction of flexible rotor blades and towers, and is typically used with tapered beam elements to build up this type of beam-like structure. The existing PART element in ADAMS already allows for direct definition of all mass moments of inertia, along with total mass and location of the center of gravity. As such, it is clearly capable of modeling a body in which the section properties vary linearly along the length.

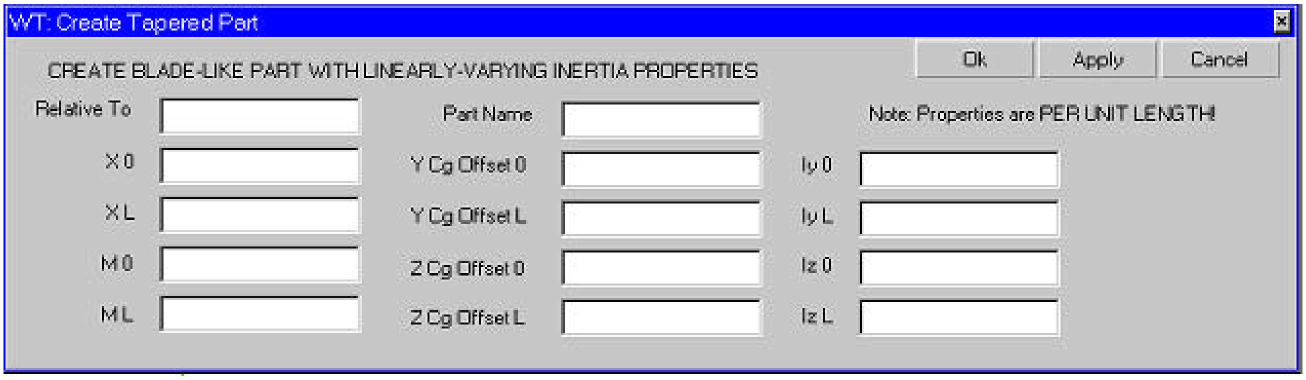
To determine the correct values for the PART inertias, the appropriate inertia integrals were evaluated symbolically, based on the assumption of linearly-varying section properties. The definitions of the inertia terms were then coded directly into the ADAMS/WT macro which builds the tapered part. A complete explanation and derivation of the tapered part inertias can be found in Appendix E.

Note that in this element, it is the cross-section mass moments of inertia per unit length, not area moments, that are assumed to vary linearly over the part's length. As can be seen in the entry panel, the tapered part is defined with respect to some particular reference coordinate system. The taper runs straight along the x axis of this system.



Like the tapered beam, tapered part elements are normally generated automatically by ADAMS/WT during the creation of flexible rotor blades or towers. Also like the tapered beam, it is possible to manually create a tapered beam element through the WT-Specific Extras menu shown above

The data entry panel for the tapered part is shown below:



Relative to

A valid ADAMS/View coordinate system definer (PART or MARKER). The tapered part will lie along its x-axis.

x_0

Location of the inboard end of the element, m.

x_L

Location of the outboard end of the element, m.

m_0

Mass per unit length at inboard end, kg/m.

m_L

Mass per unit length at outboard end, kg/m.

Part Name

ADAMS/View name to be given to this PART. ADAMS/WT will automatically name parts when they are generated as part of an aggregate element.

y_CG offset_0

Y-axis distance from reference axis to section center-of-gravity at inboard end, m.

y_CG offset_L

Y-axis distance from reference axis to section center-of-gravity at outboard end, m.

z_CG offset_0

Z-axis distance from reference axis to section center-of-gravity at inboard end, m.

z_CG offset_L

Z-axis distance from reference axis to section center-of-gravity at outboard end, m.

Iy_0

Section second mass moment of inertia about y at inboard end, kg-m.

Iy_L

Section second mass moment of inertia about y at outboard end, kg-m.

Iz_0

Section second mass moment of inertia about z at inboard end, kg-m.

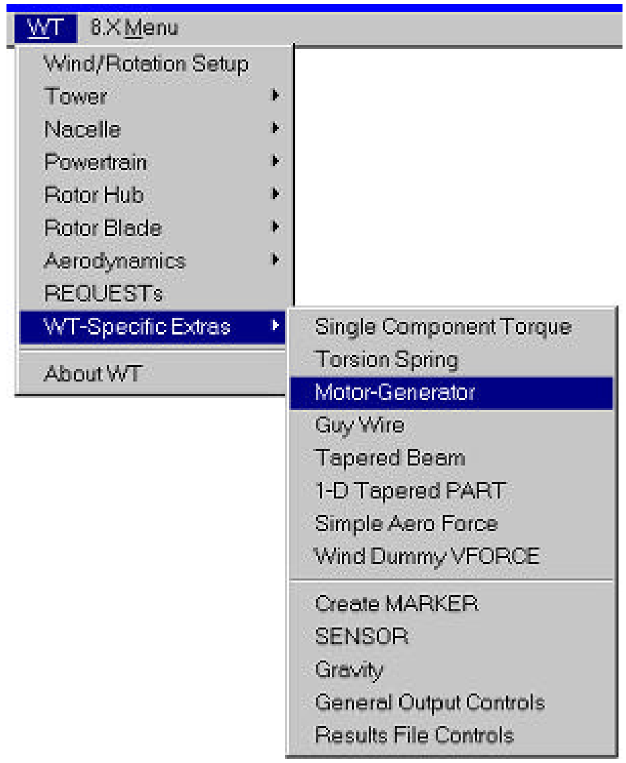
Iz_L

Section second mass moment of inertia about z at outboard end, kg-m.

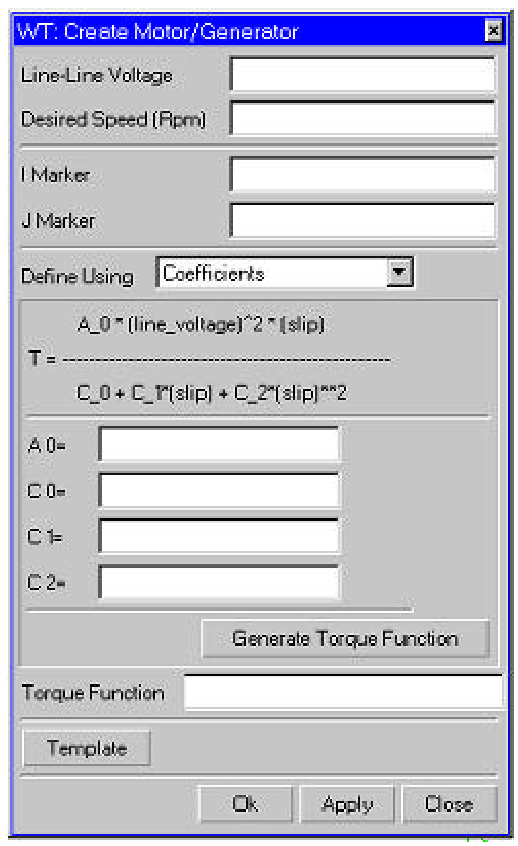
Again, after the user enters all the data and executes the panel (by clicking on OK), ADAMS/WT automatically runs the macro code which computes the inertias and CG location, and then generates the desired PART definition.

4.3 Motor-Generator

The motor-generator element in ADAMS/WT is designed to approximate the response of an induction generator. The motor-generator produces a pure action-reaction torque about the z-axes of two ADAMS MARKERs, based on a particular torque-speed-voltage relation. As normally incorporated in the power train aggregate element (see 5.4 below), one of these MARKERs will be on the generator's *stator* PART and the other on the high speed shaft. It is also possible to access the motor-generator panel through the WT-Specific Extras menu:



What appears as a motor-generator element in ADAMS/WT preprocessing is actually implemented in the model and dataset as a rotational SFORCE, which is the single-component torque element in ADAMS, with a specially-defined torque function. The data entry panel for the motor-generator is shown here:



Line-Line Voltage

The ADAMS/Solver function expression giving the applied voltage for the motor-generator, volts. Line voltage may be a constant, a function of time, or even a complex function of other system variables. The voltage will appear in the model as an ADAMS/Solver VARIABLE named *line_volts*.

Desired Speed

The ADAMS/Solver function expression giving the desired output shaft speed, revolutions per minute. The desired speed expression may also be a constant, a function of time, or function of other system variables. The desired speed will be converted automatically by ADAMS/WT into radians/second and will appear in the model as an ADAMS/Solver VARIABLE named omega_desired.

I Marker

The ADAMS/View name of the action MARKER for the motor-generator, normally on the high-speed shaft PART.

J Marker

The ADAMS/View name of the reaction MARKER for the motor-generator, normally on the stator PART.

Torque Function

The ADAMS/Solver function expression defining the motor-generator torque as a function of the line voltage in volts and the rotational speed in radians/second. The user can manually insert an expression here or ADAMS/WT can automatically generate an expression based on either Thevenin's equation or 3-D spline data (see DEFINE button below). The torque is defined positive in the motor sense, so that the motor-generator on a normally operating turbine would produce negative torque.

Define Using [Coefficients/Spline]

ADAMS/WT can automatically generate an expression based on either Thevenin's equation or 3-D spline data (see DEFINE buttons below). Note that the torque is defined positive in the motor sense, so that the motor-generator on a normally operating turbine would produce negative torque.

By default, this panel comes up in the Thevenin's mode, where you define the four coefficients in Thevenin's standard relation (shown) for motor torque as a function of voltage and slip. Slip = $1-|\Omega_{actual}/\Omega_{desired}|$ and will appear in the model as an ADAMS/ Solver VARIABLE called *slip*. A description of Thevinin's equation and the derviation of this form appears in Appendix I.

Switching the panel to the SPLINE mode allows you to input the name of the SPLINE element which gives torque as a function of slip and voltage. When defining a SPLINE for use with the motor-generator, is it imperative that the element be arranged with the torque as the y-variable, the slip as the x-variable and the voltage as the z-variable. To create a SPLINE for the motor-generator, use the standard View Build menu selecting Data Element/Spline.

Generate Torque Function

After the user fills in the four coefficients or the SPLINE name and selects this button, ADAMS/WT will automatically generate the correct expression for the motor-generator torque function and load it into the Torque Function window on the creation panel.

Template

Selecting this button will display the powertrain template for reference.

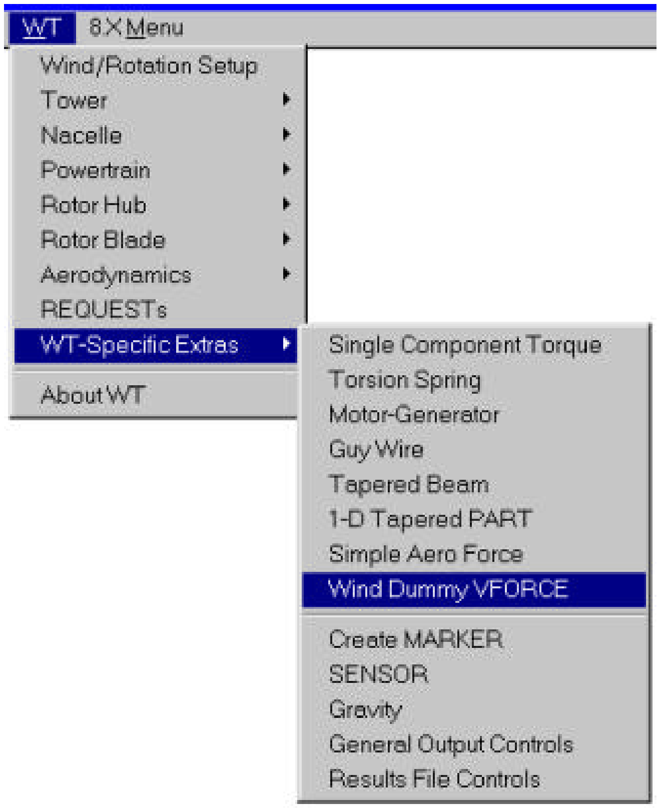
When all the data are entered and the panel is executed (OK or APPLY), WT will construct an appropriate SFORCE named *mot_gen* and put it in the model as specified in the panel. The sense of the force will be correct for the direction of rotation if you have specified it previously (using WIND/ROTATION Setup).

4.4 Wind

For a wind turbine model, the effect of ambient wind can be included in many different ways. Most useful methods are designed to take advantage of ADAMS' innate ability to resolve a vector quantity into any desired reference frame. The approach that is used in ADAMS/WT for the simple aerodynamic force (see 4.5 below) is to create a dummy vector force (VFORCE) on a dummy PART and make the global x, y and z components of that force equal to the wind velocity components. During computation of the airloads, the subroutines query ADAMS for that force vector, and resolve it into the local coordinate system so that it becomes part of the aerodynamic section's total incident velocity vector. A similar approach can be used for inflow effects (see section 6.3), but using the hub or nacelle coordinate system to define the components, instead of a dummy one.

Note that all external flow effects are already contained in the advanced airloads computations done by the AeroDyn subroutines. No wind element is needed if only the AeroDyn subroutines will be used. You can safely plan to use both types of aerodynamics, however, since including a wind element in the model will have no effect if the AeroDyn force calculations are chosen

Otherwise, the wind is not automatically created by ADAMS/WT. The user must access the wind creation panel through the WT-Specific Extras menu:



This brings up the wind creation panel:

Dummy Part

The ADAMS/View name of a PART to which the wind VFORCE will be attached. This part is usually connected to the ground via a fixed JOINT and can be used for other "dummy" elements also. See DUMMY PART button below.

Direction Marker

The ADAMS/View name of a MARKER on the dummy PART which is used as the reference for defining the wind velocity components. See DUMMY PART button below. Selecting the MARKER button will bring up the standard ADAMS/View MARKER creation panel.

Read YAWDYN.IPT File?

Whether or not to read the *yawdyn.ipt* file to find wind velocity data. This is not functional in version 2.0 of ADAMS/WT and the user should always enter *no*. The wind components must be manually defined via the standard ADAMS/View MODIFY panel for VFORCEs.

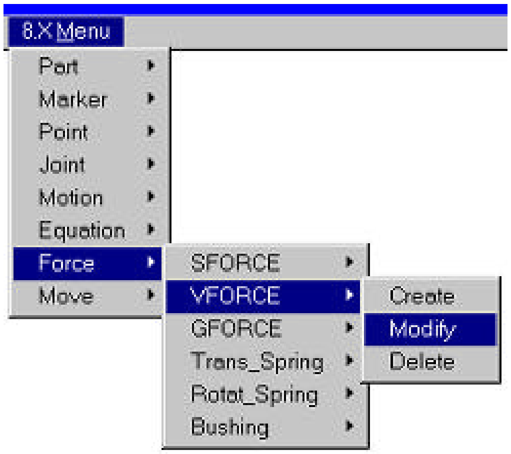
DUMMY PART

Selecting this button is part of the normal way to create wind in ADAMS/WT. Doing so displays the panel shown here:

Note that to maintain consistency with the simple aerodynamic force elements, the user must accept the default values which ADAMS/WT automatically loads into the panel windows. Executing this panel will create the dummy PART named *wind_dummy_part*, the direction definition MARKER *wind_dirs* and the fixed JOINT to lock the dummy part to the ground. It will also return you to the wind creation panel with the PART name loaded into the appropriate window.

When the wind panel is executed, ADAMS/WT creates the dummy VFORCE element *wind* and changes the color of all the associated elements to dark gray so as not to visually distract from the "real" parts of the model. The user may prefer to simply turn off the visibility of wind elements.

The automatically-created *wind* VFORCE by default has all zero components. The user should add wind to the model by modifying the *wind* using the standard ADAMS/View commands. The VFORCE modify panel is now accessible from the 8.X Menu menu. It is possible to use the separate FX, FY and FZ functions to define the wind components, or to define them in a user-written VFOSUB subroutine, if desired.



4.5 Simple Aerodynamic Force

The simple aerodynamic force element in ADAMS/WT was developed as an alternative to using the complete, nonlinear, unsteady aerodynamics in the AeroDyn subroutine package from the University of Utah. It models 2-D, steady, piecewise linear aerodynamics as expressed here:

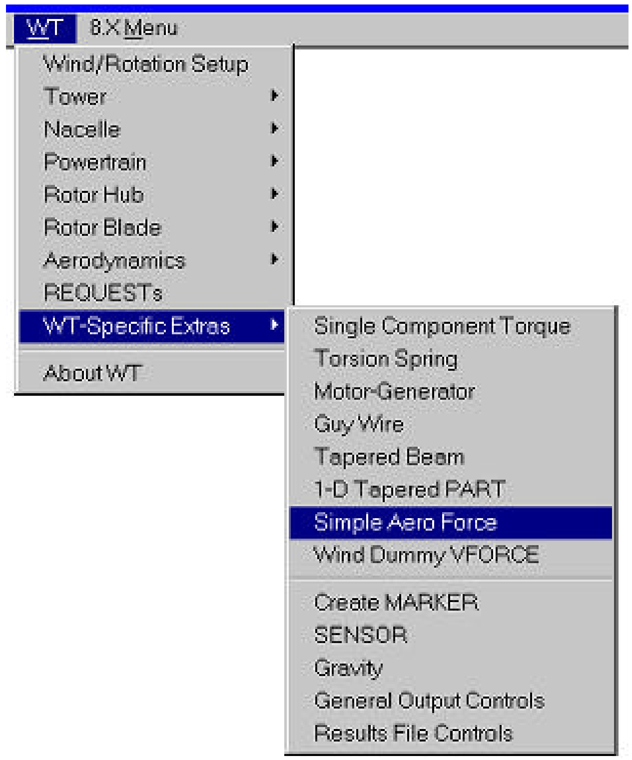
$C_L = (dC_L/d\alpha)\ \alpha$, but $|C_{Lmax}| \leq 2.0$

$C_D = C_{D0}$

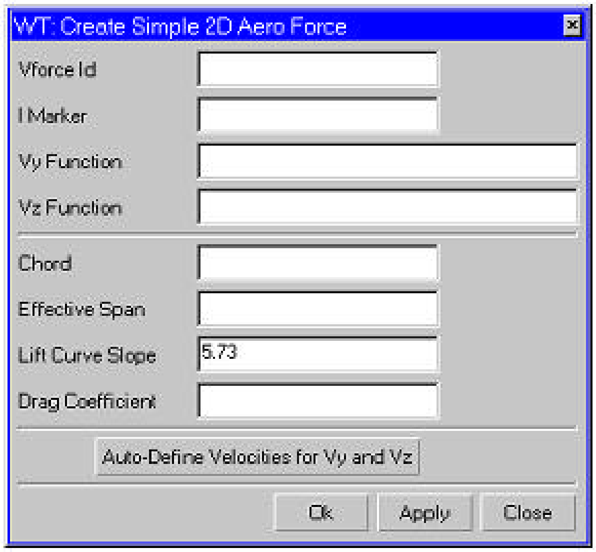
$C_M = 0$ (about the aerodynamic center)

where the angle-of-attack, α, is determined by the ratio of the incident velocity components.

Normally, either the simple aerodynamic forces or the AeroDyn forces will be automatically added to each entire rotor blades *en mass* during the system assembly process (sections 6.1 and 6.2). It is also *possible* to manually create simple aerodynamic force elements through the WT-Specific Extras menu:



Aerodynamic forces, both simple and AeroDyn types, are implemented in ADAMS/WT as multiple action-only VFORCEs attached to the various aerodynamic center MARKERs along the blades. When created automatically by ADAMS/WT, they are numbered in accordance with AeroDyn conventions. Users should modify this numbering only if they are intimately familiar with the AeroDyn code. The simple aerodynamic force creation panel is shown below:



VFORCE ID

The ADAMS/Solver identifier for this VFORCE. ADAMS/WT will name the element *saf_ID*.

I Marker

The ADAMS/View name of the MARKER on which the aerodynamic forces will act and which defines the reference axes for resolving both incident velocities and forces. The x-axis should be radial, with the y-axis toward the pressure side and the z-axis toward the leading edge.

Vy Function

The ADAMS/Solver function expression which gives the normal component of the incident velocity, positive toward the pressure side, m/sec. The user may explicitly define this function or have ADAMS/WT do it automatically by selecting the Velocities for WT button (see below).

Vz Function

The ADAMS/Solver function expression which gives the chordwise component of the incident velocity, positive toward the leading edge, m/sec. The user may explicitly define this function or have ADAMS/WT do it automatically by selecting the Velocities-for-WT button (see below).

Chord

The effective chord length for this force, m.

Effective Span

The effective span length for this force, m.

Lift Curve Slope

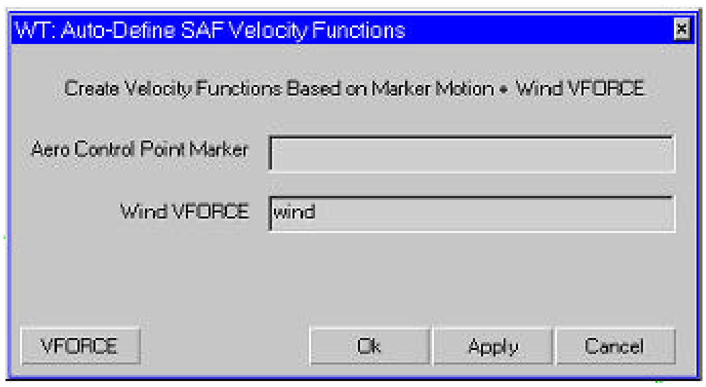
The rate of change of section lift coefficient with angle-of-attack, per radian.

Drag Coefficient

The assumed-constant section drag coefficient.

Velocities for WT

Selecting this button brings up a secondary panel for defining the incident velocity functions based solely on a wind dummy VFORCE (usually *wind*) and the global (inertial) velocity of the aerodynamic control point MARKER. The panel, whose entries are self-explanatory, is shown here:



When this panel is executed (OK or APPLY), ADAMS/WT will generate the correct function expressions and load back them into the windows in the simple aerodynamic force creation panel.

When the main simple aerodynamic force panel is executed, ADAMS/WT creates two VARIABLEs named *Vy_ID* and *Vz_ID* along with the actual VFORCE, *saf_ID*.

The VFORCE is defined specifically so that the user-written subroutine *safsub.f* can compute the airload components. The VFORCE definition in the dataset will be for FUNCTION=USER and the parameter list will contain the I MARKER ID, chord, span, lift curve slope, drag coefficient, Vy VARIABLE ID and Vz VARIABLE ID in that order. For more information on using user-written subroutines with ADAMS/Solver elements, see the *Using ADAMS/Solver Subroutines*[5].

5. *Using ADAMS/Solver Subroutines*, Version 9, PN 91SOLVSUB-01, Mechanical Dynamics, Ann Arbor, MI,1998.